UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/675,921	09/29/2000	Andrew Harvey	50325-0126	8996

29989	7590	05/31/2005

HICKMAN PALERMO TRUONG & BECKER, LLP
2055 GATEWAY PLACE
SUITE 550
SAN JOSE, CA 95110

EXAMINER
NGUYEN, QUANG N

ART UNIT	PAPER NUMBER
2141	

DATE MAILED: 05/31/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

	Application No.	Applicant(s)
Office Action Summary	09/675,921	HARVEY ET AL.
	Examiner	Art Unit	
	Quang N. Nguyen	2141	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *3/25/05*.

2a)☒ This action is **FINAL**.   2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-33* is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-33* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on <u>29 September 2000</u> is/are: a)☒ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All   b)☐ Some * c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☒ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.

4)☐ Interview Summary (PTO-413) Paper No(s)/Mail Date _____.

5)☐ Notice of Informal Patent Application (PTO-152)

6)☐ Other: _____.

Detailed Action

1.   This Office Action is in response to the Amendment filed on 03/25/2005. Claims

1-2, 4-6, 12-14, 17-18, 22-28 and 30 have been amended. Claims 1-33 are presented

for examination.

Claim Rejections - 35 USC § 103

2.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set
> forth in section 102 of this title, if the differences between the subject matter sought to be patented and
> the prior art are such that the subject matter as a whole would have been obvious at the time the
> invention was made to a person having ordinary skill in the art to which said subject matter pertains.
> Patentability shall not be negatived by the manner in which the invention was made.

3.   **Claims 1-3 and 8-14 are rejected under 35 U.S.C. 103(a) as being**

unpatentable over Chen et al. (US 6,195,694), herein after referred as Chen, in

view of Royal, Jr. et al. (US 6,571,201), herein after referred as Royal, and further

in view of Malik et al. (5,832,503), herein after referred as Malik.

4.   As to claim 1, Chen teaches a method of automatically configuring a network

device, comprising:

receiving a request from the network device to provide configuration information

(i.e., server 195 receiving a request from the browser 160 via the network interface 150

for the application files/configuration sets 175) (Chen, C8: L31-36);

retrieving a template *(i.e., HTML file 500 as illustrated in Fig. 5)* describing a

device configuration (Chen, Fig. 5 and C9: L17-43);

However, Chen does not explicitly teach the retrieved template comprises

symbolic references to one or more parameters that may receive values specific to a

particular device; retrieving, based on the symbolic references, one or more values of

parameters specific to the network device; creating and storing a device-specific

instance of the configuration information based on the template and the values of

parameters; and said configuration information conforming to an Extensible Markup

Language Document Type Definition (XML DTD) and comprising one or more XML tags

that delimit a beginning and an ending of the configuration information.

In a related art, Malik teaches a method and apparatus for configuration

management in communications networks, wherein a configuration manager 18

retrieves a template *(i.e., retrieves a list of attributes for a device of a certain model type*

as item 40 in Fig. 3); obtains the values of certain attributes *(i.e., obtains data which*

define the characteristics of the network device being modeled as item 42 in Fig. 3) and

the resulting configuration created with the template containing the attributes and values

maybe stored in the configuration manager, in another storage device, or in the

SPECTRUM database (Malik, C3: L13-46).

In another related art, Royal teaches a method of automatically configuring a network device *(a fuel dispenser 110)* via XML-based data exchanges with the remote system 130 and site controller 120, wherein the XML-formatted data comprises one or more "elements" delimited by a start tag, an end tag and intervening data (Royal, Fig. 2A, C4: L46-48 and C5: L35-44).

Therefore, it would have been obvious to one having ordinary skills in the art at the time the invention was made to combine the teachings of Chen, Malik and Royal to retrieve a template comprises symbolic references to one or more parameters that may receive values specific to a particular device; retrieve one or more values of parameters specific to the network device; create and store a device-specific instance of the configuration information based on the template and the values of parameters; and conform configuration information to an XML DTD comprising a beginning and an ending tag to delimit the configuration information since such methods were conventionally employed in the art to allow a configuration manager processing and storing model-based configuration data, utilizing templates for configuring a plurality of network devices in order to make such configuration management less time-consuming, expensive and error prone (Malik, C1: L6-12), wherein XML supports a richer set of document elements and applies better to various publishing media to allow a remote system to conveniently retrieve, monitor, or update/configure network devices using data items tagged in accordance with the defined grammar (Royal, C3: L17-27).

5.   As to claims 2-3, Chen-Malik-Royal teaches the method of claim 1, further comprising the steps of:

testing the device-specific instance of configuration information to determine whether it is well-formed with respect to the XML DTD *(i.e., stored data retrieved is converted into XML-formatted data using an XML processor to ensure that it is well-formed with respect to the XML DTD)* (Royal, C8: L7-40);

providing the device-specific instance of configuration information to the network device over a reliable transport protocol *(i.e., XML configuration files maybe transferred between the fuel dispenser 110 and the remote system 130 using several known techniques such as HTTP for URL named request response files transfers or FTP via the Internet)* (Royal, C7: L1-20), wherein the network device ensures that all of the configuration information is received by checking the one or more XML tags that delimit a beginning and an ending of configuration information *(wherein XML-formatted data comprises a start tag and an end tag delimiting intervening data for example, passing the "addresses" structure to extract individual "address" entries simply entails parsing the overall data set based on the start and end tags)* (Royal, C4: L47-67 and C5: L1-5).

6.   As to claim 8, Chen-Malik-Royal teaches the method of claim 1, further comprising the steps of:

applying the device-specific instance of configuration information to the network device *(i.e., executing configuration file 500 line by line to invoke APIs to configure the device)* (Chen, C8: L1-13 and L31-54);

receiving a user request to cancel application of the configuration information

(i.e., user actions can change which application files 175, files 500 are executed and/or

whether or not some of the application files are executed) (Chen, C8: L50-54).

restoring the network device to its state prior to application of the device-specific

instance of configuration information *(i.e., a "safe load" option helps to control the load*

process, if the load fails, the configuration manager tries to restore, i.e., roll back the

original configuration to the model) (Malik, C9: L1-6).

7.   As to claims 9-11, Chen-Malik-Royal teaches the method of claim 1, comprising

the step of receiving an HTTP request that identifies an Active Server Page *(ActiveX*

technology) or Java Servlet of a configuration service *(the embedded programs can be*

implemented using Java script, and/or a Java applet and/or any other embedded

program which uses plug-ins) (Chen, C9: L33-35 and C19: L62-67) that can provide

configuration information and that includes a unique identifier of the network device

(Malik teaches the configuration manager is capable of capturing attribute values and

instance ID, i.e., unique identifier, wherein model-specific configurations are restricted to

one device) (Malik, C7:L58 – C8:L8).

8.   As to claims 12-13, Chen-Malik-Royal teaches the method of claim 1, further

comprising the additional steps of:

retrieving a reference to a template, which comprises symbolic references to one

or more parameters that may receives specific values to a particular device, describing

the configuration information from a directory service *(Malik teaches selecting a template, which contains the attributes and values as item 42 in Fig. 3, from a database of a configuration server, hence, it is inherent that a reference to the location of templates such as memory address, database location, directory, subdirectory, etc., was received)*; and·

retrieving a container object associated with the network device from a directory in a directory service and obtaining the values of parameters from directory objects contained within the container object *(Malik also teaches the configuration manager retrieving a model type having an associated set of attributes, i.e., a container object, associated with the network device and retrieving the values for each of the attributes from the attributes values in the databases, i.e., values of parameters, used to configure the device)* (Malik, Fig. 5, C2: L14-30 and C7: L24-32).

9.   As to claim 14, Chen-Malik-Royal teaches the method of claim 5, wherein the step of syntax checking additionally comprises parsing one or more configuration commands from the device-specific instance of configuration information using a parser of an operating system that is executed by the network device *(XML tags included within the XML-formatted data allow either the fuel dispenser 110, i.e., the network device, or the remote system 130 to easily parse the received data, using an XML processor 206 and/or standards-based XML/HTML compliant software, which could be an operating system executed on the remote system 130)* (Royal, C6: L49-58 and C8: L44-50).

10.   Claims 4-5 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chen-Royal-Malik and further in view of Mattox (US 5,404,321).

11.   As to claims 4-5, Chen-Malik-Royal teaches the method of claim 2, further comprising the steps of:

at the network device, applying the device-specific instance of configuration information to the network device (*i.e., executing configuration file 500 line by line to invoke APIs to configure the device*) (Chen, C8: L1-13 and L31-54);

when a syntax error is detected during the syntax checking step, publishing an event that reports the syntax error using an event service (*Malik teaches generating alarms to an event log to indicate whether or not configuration was successful*) (Malik, C2: L33-36 and C9: L48-61).

However, Chen-Royal does not explicitly teach, at the network device, syntax checking only configuration commands of the device-specific instance of the configuration information to determine whether the configuration commands therein conform to a command language that is understood by the network device.

In a related art, Mattox teaches the system initialization routine wherein the DOS loads the computer program such as a device driver based on a "DEVICE" command in the configuration file (*or a program such as XML parser, syntax checker, etc., could be loaded*) to check the configuration commands for syntax errors and sets various internal variables in the program according to the commands (Mattox, C9: L3-13).

Therefore, it would have been obvious to one having ordinary skills in the art at the time the invention was made to modify the teachings of Chen-Royal-Malik and Mattox to include syntax checking configuration commands of the device-specific instance of the configuration information to determine whether the configuration commands therein conform to a command language that is understood by the network device and publishing an event that reports the syntax error using an event service since such methods were conventionally employed in the art to avoid the execution of code that is not syntactically correct that can provide results, which are unpredictable and detrimental, and to allow user to view the results of the configuration and make the decisions based on the provided results.

12.   Claims 6-7 and 15-16 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chen-Royal-Malik and further in view of Suarez (US 5,790,789).

13.   As to claims 6-7, Chen-Malik-Royal teaches the method of claims 1 and 5, but does not explicitly teach the additional step of generating an event to an event service to which the plurality of network devices subscribe, wherein the event announces that the configuration commands conform to a correct syntax.

In the related art, Suarez teaches that event services can be used to provide the ability to create, update, publish and subscribe to global or system defined events by constantly monitoring environments and reacting accordingly to allow agents, services and users to define reactions to certain events (Suarez, C21: L35-50).

Therefore, it would have been obvious to one having ordinary skills in the art at the time the invention was made to combine the teachings of Chen-Malik-Royal and Suarez to generate an event to an event service announcing that the configuration commands conform to correct syntax since such methods were conventionally employed in the art to allow the testing of one device before applying the configuration to many to make sure that only one device has the chance of entering an error state instead of the entire network if the configuration commands are faulty.

14.   As to claims 15-16, Chen-Royal-Malik-Suarez teaches the method of claim 1, further comprising the additional steps of:

determining that a partial configuration should be sent to one or more network devices *(determining what attributes of a model type are of interest for configuring the device, i.e., partial configuration)* (Malik, Fig. 6 and C7: L8-15);

publishing the partial configuration trigger event to an event service (Suarez, C21: L35-50); and

providing the partial configuration to one or more network devices (Malik, C2: L18-26).

15.   Claims 17-31 recite limitations similar to the limitations of claims 1-16; therefore, they are rejected under the same rationale.

16.   **Claims 32-33 are rejected under 35 U.S.C. 103(a) as being unpatentable over Pan et al. (US 6,775,701), herein after referred as Pan, in view of Malik et al. (5,832,503), herein after referred as Malik.**

17.   As to claim 32, Pan teaches a method and system for reserving a network resource, comprising:

receiving a request for network topology information from the computer program application *(i.e., network resource manager 11 receives a "service reservation" from a service agent 12 running on a network device 14)* (Pan, C2: L48-50);

resolving elements of the topology into application-specific values *(i.e., network topology monitor 24 obtains and validates a path for the requested service, gathers information such as maximum bandwidth per link, filters, flow identifiers, CoS parameters on the network device, and/or any items that are required to implement the service)* (Pan, C6:L57 – C7:L46 and C9: L8-11), resulting in creating and storing resolved topology information *(the derived topology information is mapped to links and is used to update the dynamic topology map stored in repository 25)*; (Pan, C6: L19-21, C8: L1-3 and L15-20);

providing the resolved network topology information to a configuration agent within the application that is configured to re-configure the computer program application to operate with the resolved network topology *(i.e., after network topology monitor 24 obtains a path for the requested service, engine 26 instructs mechanism*

adapter 19 to communicate the required information to network device 14 to implement the service) (Pan, C9: L1-11).

However, Pan does not explicitly teach retrieving a template of network topology information from a repository.

In the related art, Malik teaches selecting/retrieving a template, associated with the network device and then using the template as an index to retrieve attribute values *(i.e., application-specific values)*, used to configure the network device (Malik, Fig. 5, C2: L14-30 and C7: L24-32).

Therefore, it would have been obvious to one having ordinary skills in the art at the time the invention was made to combine the teachings of Pan and Malik to include a template that can be filled with application-specific values for configuration management in communication networks because templates provide a level of object-oriented topology to the network, wherein a set of network devices can be associated with one single model type, thus simplifying configuration (Malik, C2: L36-50).

18.   As to claim 33, Pan-Malik teaches the method of claim 32, but does not explicitly teach application-specific syntax checking of elements of the template.

"Official Notice" is taken that both the concepts and advantages of checking and ensuring program code *(i.e., checking and ensuring elements of the template)* is syntactically correct before executing the code are well known and expected in the art.

Therefore, it would have been obvious to one having ordinary skills in the art at the time the invention was made to modify the teachings of Pan and Malik to include

syntax checking since such methods were conventionally employed in the art to avoid

the execution of code that is not syntactically correct that can provide results which are

unpredictable and detrimental.

Response to Arguments

19.   In the remarks, applicant argued in substance that

(A)   Prior Art do not teach or suggest, "a template with symbolic references to

parameters, much less retrieving values of parameters specific to the network device

based on the symbolic references", as featured in claim 1.

As to point (A), **Malik** teaches a method and apparatus for configuration

management in communications networks, wherein a configuration manager 18

retrieves a template containing a list of attributes for a device of a certain model type as

item 40 illustrated in Fig. 3 *(i.e., retrieving a template with symbolic references to*

parameters); then the configuration manager 18 captures the values of the attributes

listed in the template *(i.e., retrieving values of parameters specific to the network device*

based on the symbolic references), by retrieving the values from the SPECTRUM

model, i.e., obtains data which define the characteristics of the network device being

modeled as item 42 illustrated in Fig. 3 **(Malik, C2: L16-23 and C3: L13-46)**.

(B)   Prior Arts do not teach or suggest, "the request from the network device includes a unique identifier of the network device", as recited in claims 9-11 and 19-21.

As to point (B), **Chen-Malik-Royal** teaches the method of claim 1, comprising the step of receiving an HTTP request that identifies an Active Server Page *(ActiveX technology)* or Java Servlet of a configuration service *(the embedded programs can be implemented using Java script, and/or a Java applet and/or any other embedded program which uses plug-ins)* (**Chen**, C9: L33-35 and C19: L62-67) that can provide configuration information and that includes a unique identifier of the network device *(Malik teaches the configuration manager is capable of retrieving a template containing a list of attributes of a device or a certain model type and capturing the values of the attributes listed in the template and instance IDs, i.e., unique identifiers if any such as AT_NET_ADDR, AT_PHYS_ADDR, DEVICE_NAME, SYSNAME, wherein model-specific configurations are restricted to one device)* (**Malik**, C7:L58 – C8:L8).

Also, to one of ordinary skill in the art, for every HTTP request, it is necessary *(inherently)* to include a source address *(i.e., a unique identifier identifying the network device)*, a destination address, a source port, and a destination port in the header portion of the request.

(C)   Prior Arts do not teach or suggest, "retrieving a container object associated with the network device from a directory in a directory service and obtaining the values of parameters from directory objects contained within the container object", as recited in claim 12-13, 22, 25 and 31.

As to point (C), **Malik** also teaches the configuration manager retrieving a template/model type having an associated set of attributes *(i.e., a container object)* associated with the network device from the database of a configuration server *(here, it is inherent that a reference to the location of templates such as memory address, database location, directory, subdirectory, etc., was received)* and retrieving the values for each of the attributes from the attributes values in the databases, i.e., values of parameters, used to configure the device (**Malik**, Fig. 5, C2: L14-30 and C7: L24-32).

(D)   Prior Arts do not teach or suggest, "any type of request for network topology information", as recited in claim 32.

As to point (D), **Pan** teaches when resource validator 22 receives the new "service reservations" and passes source and destination network addresses from the service reservations to network topology monitor 24. Network topology 24 obtains a path for the requested service *(i.e., request for network topology information)* and then provides that path to the resource validator 22 (**Pan**, C6: L54 – 67 and C7: L1-9).

20.   Applicant's arguments as well as request for reconsideration filed on 03/25/2005

have been fully considered but they are moot in view of the new ground(s) of rejection

and not deemed to be persuasive.

21.   **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time

policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE

MONTHS from the mailing date of this action. In the event a first reply is filed within

TWO MONTHS of the mailing date of this final action and the advisory action is not

mailed until after the end of the THREE-MONTH shortened statutory period, then the

shortened statutory period will expire on the date the advisory action is mailed, and any

extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of

the advisory action. In no event, however, will the statutory period for reply expire later

than SIX MONTHS from the mailing date of this final action.

22.   Further references of interest are cited on Form PTO-892, which is an

attachment to this office action.

23.   Any inquiry concerning this communication or earlier communications from the examiner should be directed to Quang N. Nguyen whose telephone number is (571) 272-3886.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's SPE, Rupal Dharia, can be reached at (571) 272-3880. The fax phone number for the organization is (703) 872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

RUPAL DHARIA
SUPERVISORY PATENT EXAMINER